

ABSTRACT

A b-tree may contain multiple keys. Each key may contain multiple column values of different types. In order to allow quick byte by byte comparison of the keys, a normalized form of the keys may be determined by transforming each column value, through a type specific transformative function, into a normalized form. This normalized form allows keys to be quickly compared and may be used in compressing the underlying b-tree. Each normalized key is compressed relative to the previous key by replacing the common prefix bytes with a common prefix length byte containing the number of bytes in common between the key and the previous key. The compressed keys are stored sequentially in memory with no gaps in between.